

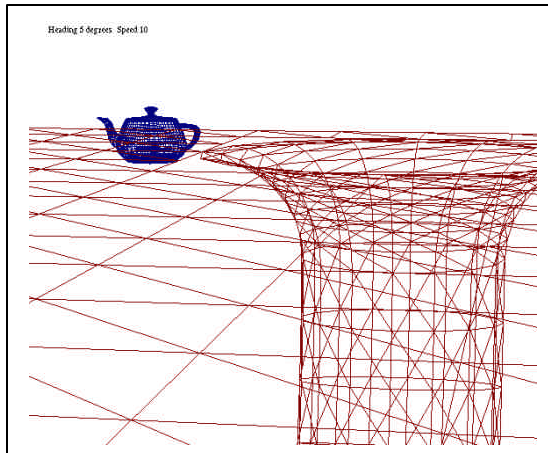
# NKU CSC 480

## Spring 2006

### Programming Assignment 3

---

For this assignment you will write a function that allows the user to pilot a craft along a plane torn by a wormhole. The wormhole should be drawn in wire frame, as in Assignment 1, but you may wish to tweak it a little to make it more visually interesting. Sprinkle the plane with a few small wire frame objects, such as teapots and dodecahedrons of various colors. The craft hovers a small distance above the plane, allowing a perspective view roughly like the one shown below. If the craft moves inside the outer radius of a wormhole, the craft is sucked through the wormhole at a constant speed and emerges out the other end approximately 5 seconds later.



You should encapsulate the craft in a class named **SpaceCraft**. It will have mutators that change its speed and heading, and will also encapsulate the camera-- the parameters to **gluLookAt()** and **gluPerspective()**.

In a later assignment we will add lighting and textures and independently moving objects to the scene, as well as more interesting dynamics.

When the program launches, the user sees a first person view from the craft, which is sitting on the middle of the plane with zero velocity. Across the top of the viewport the user should read the current speed and the heading (0 to 360 degrees). The user interacts with the program through the following controls:

↑,↓ keys	add/subtract a small increment from the speed (negative speed means move in reverse)
←,→ keys	add a small left/right increment to the heading angle
[space bar]	toggle the animation (paused/running)
v	toggle between first person and third person POV
r	resets the craft to its initial speed and position

In the third person point of view, the plane and wormhole are rendered in stationary perspective from a short distance away, with the craft rendered as a distinctly colored small cone.

Other requirements:

- When the craft reaches the boundary of the plane, it cannot go further. You can either set its speed to zero or, if you like, make it ricochet.
- Add some sound effects by playing some **.wav** files using `PlaySound()` when certain events happen, such as hitting a boundary or being sucked into the wormhole.

**DUE:** By 9am Tuesday **March 21**. Submit your program in a single file named **Pilot\_\_\*.cpp**, with your last name replacing \*. No late assignments will be accepted. As always, programs are graded on design and style as well as nominal correctness and visual aesthetics. Start early, work incrementally, test as you go.

## CSC 480 Spring 2006

### Program 2: Wormhole Pilot – Evaluation Sheet

#### Correctness (20)

- \_\_\_ /3 plane, wormhole (torus + “adaptors”), and objects drawn correctly
- \_\_\_ /3 first person perspective correct
- \_\_\_ /3 third person perspective correct
- \_\_\_ /3 velocity and heading controls correct
- \_\_\_ /3 behavior near/in wormhole correct
- \_\_\_ /1 reset function works
- \_\_\_ /1 animation pause works

#### Design / Style (5)

- \_\_\_ / 3 program demonstrates good visual and auditory interest, is easy to use
- \_\_\_ / 3 appropriate modularity and efficiency in code
- \_\_\_ / 2 clarity: appropriate commenting, layout, naming

**Total:** \_\_\_ / 25

**Notes:**